



Awareness, Usage and Discovery of Swipe-revealed Hidden Widgets in iOS

Nicole Ke Cheng Pong, Sylvain Malacria

► To cite this version:

Nicole Ke Cheng Pong, Sylvain Malacria. Awareness, Usage and Discovery of Swipe-revealed Hidden Widgets in iOS. ISS 2019 - Proceedings of the ACM Conference on Interactive Surfaces and Spaces, Nov 2019, Daejeon, South Korea. hal-02293836

HAL Id: hal-02293836

<https://inria.hal.science/hal-02293836>

Submitted on 22 Sep 2019

HAL is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

Awareness, Usage and Discovery of Swipe-revealed Hidden Widgets in iOS

Nicole Ke Chen Pong^{1,2} Sylvain Malacria^{1,2}

¹Inria, France

²Univ. Lille, UMR 9189 – CRISTAL, Lille, France
nicole.pong@inria.fr, sylvain.malacria@inria.fr

ABSTRACT

Revealing a hidden widget with a dedicated sliding gesture is a common interaction design in today's handheld devices. Such "Swhidgets" (for swipe-revealed hidden widgets) provide a fast (and sometime unique) access to some commands. Interestingly, swhidgets do not follow conventional design guidelines in that they have no explicit signifiers, and users have to discover their existence before being able to use them. In this paper, we discuss the benefits of this signifierless design and investigate how iOS users deal with this type of widgets. We report on the results of a laboratory study and an online survey, investigating iOS users' experience with swhidgets. Our results suggest that swhidgets are moderately but unevenly known by participants, yet the awareness and the discovery issues of this design is worthy of further discussion.

Author Keywords

swhidgets; hidden controls; gestural interaction; iOS.

CCS Concepts

•Human-centered computing → Empirical studies in HCI; Mobile devices; Gestural input;

INTRODUCTION

Due to mobility constraints and limited interaction possibilities, the design of user interfaces in modern smartphones has evolved toward interfaces providing the minimal required amount of features and decoration. To achieve this, UI designers sometimes take extreme measures that go against the established guidelines, by removing features that may be important for users to learn about the interface itself. A typical example of such design are *Swipe-revealed hidden widgets* (that we abbreviate as *Swhidgets*), widgets that by default are entirely hidden under another interface element or the screen bezels. To use swhidgets, one first uncovers them with a simple swipe gesture relying on the metaphor of sliding physical objects, as for the iOS *Control Panel* (Figure 1-right) and message times (Figure 2-left).

Swhidgets are becoming common features in today's touch-based devices, implemented in both Apple iOS and Google Android. Interestingly though, the graphical design of swhidgets is extremely minimal and there is no signifier of any kind that might suggest to users the availability of the hidden command or help to figure out where and how to perform them. This goes against most established HCI guidelines [5, 9, 16, 20, 28, 30] advising to expose controls, use clear interface structures, and exploit perceptible *signifiers* to inform users about the presence of interactive elements, which may create usability issues [17, 21, 22].

In spite of their mainstream availability, little is known regarding swhidgets and how users actually interact with such controls. In this paper, we theorize about their design that goes against established HCI guidelines, discuss their potential benefits, and investigate how relatively tech-friendly users adopt, interact and discover *swhidgets*. First, we analyze the existing *swhidgets* of iOS, highlight their lack of signifier and discuss how users may discover them. Then, we report on the results of a laboratory study conducted with 28 participants, investigating which input methods users *in practice* know and use when prompted to perform operations that can be achieved in several ways, including with a *swhidget*. Finally, we report on the results of an online survey with 144 respondents, investigating which input methods users *report* to know and use when performing operations that can be achieved with a *swhidget*, and investigating deeper how users discover this type of controls. Our results suggest that these controls are moderately but unevenly known by users, preferred to other types of interactions when they are known, and that their signifierless design requires users to discover them through explicit exploration. Our work makes the following contributions:

- Provides empirical results on the awareness, usage and discovery of various *swhidgets* on iOS devices.
- Discusses user behaviors and expectations about *swhidgets* that we observed or that participants expressed, which may have practical value for designers.

TYPES AND ROLES OF SWHIDGETS IN IOS

As stated previously, we define *swhidgets* (for swipe-revealed hidden widgets) as widgets that are normally entirely hidden from the user, who can uncover them with a simple swipe gesture relying on the metaphor of sliding physical objects to drag the *swhidget* into view. The use of a physical drag metaphor is a distinctive feature of *swhidgets* that may provide the type of user guidance discussed by Norman as lacking

from other gestural interfaces [21]. As such, *swidgets* differ from swipe-based interactions techniques that do not rely on a sliding metaphor such as Bezel Menus or Bezel Taps [11, 27]. While *swidgets* can be found on various touch-based operating systems, we will focus on the iOS platform as this is a popular and consistent operating system without additional manufacturer overlay, relying on simple physical metaphors for interaction [2]. However, categories proposed below are still likely to apply to *swidgets* on other platforms.

Types of Swidgets in iOS

Different types of *swidgets* are available in iOS, each relying on different interface schemes that can be classified from a metaphorical perspective in the following way:

System swidgets are “pulled over” the main view from an edge of the display, following the metaphor of a roll-up projection screen. Examples of *system swidgets* are the notification center and the control panel (Figure 1), which are revealed with sliding gestures from the edge of the display – respectively downward from the top and upward from the bottom. *System swidgets* can be revealed regardless of the application/view and overlay the entire main view to provide various system commands and information.

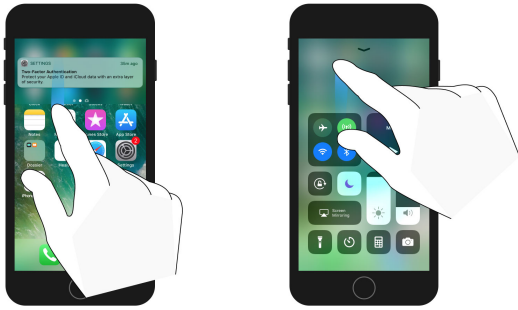


Figure 1. Examples of *system swidgets* in iOS containing various controls that can be accessed from anywhere in the OS. Left: downward swipe from top edge reveals the information/notification panel; Right: upward swipe from bottom reveals the control panel.

View swidgets are concealed on the side of a view or container, outside its boundaries, and revealed by pulling them into view, i.e. by sliding the finger in the opposite direction of the *view swidget*. Examples are the exact arrival time of SMS messages, revealed with a leftward sliding gesture from anywhere in the discussion view (Figure 2-left), or the search text field that is located at the top of most element lists (typically e-mail list in the Mail application) but hidden by default, only to be revealed when scrolling over or back to the top.

Item swidgets are hidden next to or under an object of interest within the application and provide contextual functions to interact with this object. Examples are the commands operating on e-mails in the Mail app, that can be revealed by sliding horizontally an individual e-mail (Figure 2-right). *Item swidgets* differ from *view swidgets* in the sense that their revealing gesture must be performed on a specific element.

Role of Swidgets

Swidgets are used conjointly with other types of controls in the design of iOS interfaces and cannot be understood fully if

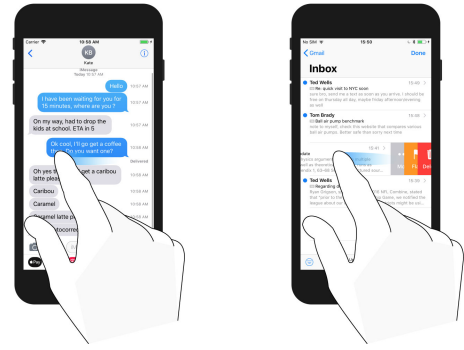


Figure 2. Examples of *view* and *item swidgets* in iOS. Left: A *view swidget* in Message, where a leftward swipe from anywhere in the discussion view reveals the exact time each message was sent; Right: A *view swidget* in Mail, where a leftward swipe on an e-mail from the list reveals a set of buttons.

taken in isolation. Most *swidgets* provide a faster access to some functionalities that can also be accessed through traditional widgets. Interestingly, some *swidgets* provide access to advanced features with no real equivalent, and where the conventional *tap* and *scroll* interaction paradigm would only provide access to limited versions of the feature. In iOS, this is typically the case of exact SMS arrival times that can only be accessed through a *view swidget*, whereas conventional interaction would only provide discussion start and resume times. Similarly, the search feature in the Mail application is not accessible through another widget than the dedicated *view swidget*¹. Depending on the *swidgets*, it can therefore be seen as a shortcut for experienced users (the default input method being the conventional controls), or as the main input method with conventional controls provided only as a fall-back for users who do not know the *swidget*. The singularity of *Swidgets* lies in the fact that they heavily rely on the metaphor of moving objects, since their purpose is to eventually bring additional widgets into view. As such, they differ from other gesture-based interaction methods (typically pinch-to-zoom, force-input or the swipe-up gesture as replacement for the home button on the recent iPhone X) not covered in this paper.

RELATED WORK

User interaction with Swidgets

To the best of our knowledge, there is very little public data on current usage of *swidgets*. The only available dataset can be found on a blogpost reporting awareness and usage proportion of a subset of system *swidgets* in iOS 11 [24], but does not accurately describe its methodology and experimental procedure, questioning the generalization of the results. As such, it remains unclear how many of these controls users actually use and how they were discovered. Indeed, academic research has mostly investigated the performance of novel edge-based interaction techniques, such as menus or keyboards [11, 25, 27] that do not rely on a physical metaphor that may help the users to uncover these features. Similarly, Schramm et al. [26] investigated the transition to expertise in *Hidden toolbars*, a name describing various types of UIs making functionality

¹The search feature in mail is also accessible through Siri, which relies on a different interaction paradigm

available only when users explicitly expose them through a dedicated interaction, such as menubars (revealing commands when the menubar is clicked), Windows 8's "charms" [10] or certain *swidgets* (typically System *swidgets*). Their work was focused on the performance of such interfaces, proposing four hidden toolbar designs (all different from actual *swidgets* designs) and comparing their performance in terms of selection time and item location learning. As a result, their work is informative regarding the performance of hidden toolbar designs they propose, but is not adequate to fully understand how users currently interact with *swidgets*.

Similarly, studies conducted on other related "hidden" or "expert" interaction techniques can help to apprehend interaction with *swidgets*. This is the case of the work conducted by Avery and Lank surveying the adoption of three multitasking multi-touch gestures on the iPad [3]. Their results show that users tend to have varying levels of knowledge and actual usage of these techniques between completely ignoring them and perfectly mastering the art of how and when to use them, and that despite being "hidden", these gesture-based techniques can get high levels of user awareness and willingness to use. Regarding keyboard shortcuts on desktop, studies conducted in Microsoft Word [13, 29] revealed that users do not systematically use the hotkeys they know, which may be explained by the fact they often underestimate their potential benefits, but not only, because they also often fail to use them when they know it will be faster, even under time pressure.

Discovery of Swidgets

Swidgets are by default "hidden" under the screen bezel, yet their existence is not signified to users, possibly as an intention to minimize visual clutter [26] and keep the UI clean in order to improve user experience and performance [19]. The iOS *Human Interface Guidelines* echo these expected benefits in the themes of *clarity* and *deference*: "Content typically fills the entire screen, [...] Minimal use of bezels, gradients, and drop shadows keep the interface light and airy, while ensuring that content is paramount." A small dash-shaped handle used to be displayed near the bottom edge of the screen when the control panel was introduced in iOS, but more recent releases show it on extraordinary occasions only (for application in full-screen mode such as games and video players), or use it as a replacement for the home button (typically on iPhone X where the home button was removed). As recently pointed by Mayer et al. [15], the swipe gestures of *swidgets* are not visually communicated to the users. Instead, device manufacturers rely on dedicated tips and animations that are either shown to the user when setting up the device, or showcased on stage when the system is presented to the tech-oriented press.

A more integrated to everyday interaction way to discover *swidgets*, is to explore the interface by performing series of inputs and expecting the system to respond to them. Schramm et al. [26] advocate that with edge-based interactions becoming more common, users might appropriate the physical metaphor of sliding object and become more likely to expect *swidgets*, thus exploring the operating system to discover them. However, Schramm et al. also acknowledge that one cannot expect novice users to guess their availability or the actions used to

access them regardless. Therefore, users still need to have discovered a first *swidget* before being able to reproduce a similar input somewhere else in the interface. In addition, it still requires users to *explicitly* explore the interface in the first place, while they maybe be too engaged in their tasks to do it, even if it would eventually improve performance [6].

Another possible way to discover *swidgets* is through accidental revelation, that is, when the *swidden* widget is revealed whereas it was not the first intention of the user. For instance, one of iOS *swidgets* that is likely to be accidentally revealed are the *Search* view *swidget* located on top of some lists, that may be brought into view when the user overshoots while scrolling back to the top of the list. Other *swidgets* such as Item *swidgets*, typically in the Mail application, are less likely to be accidentally discovered as there is no reason to expect users to perform an horizontal swipe on an item in a vertically scrollable list. Moreover, accidental activation can be confusing and users might be unable to reproduce the input operation that triggered that accidental activation.

Finally, users might discover the existence of *swidgets* from their own social network, typically through friends, colleagues or family. Indeed, impact on input mechanism adoption of witnessing others performing them has already been observed in the context of keyboard shortcuts [23].

USER STUDIES RATIONALE

Interaction with *swidgets* remains seldom studied and little is known regarding whether users are aware of *swidgets* or not, if they use them and how they discovered them. We therefore conducted two studies focusing on the iOS operating system and its most used default applications.

The first is a laboratory study conducted with 28 participants by asking them to perform specific operations (in blue in table 1) on an iOS device and observing which input modality participants spontaneously used. This study investigates whether participants are aware of and use *swidgets* without influencing their decision by asking them explicitly about this type of interfaces. The second study is an online survey conducted with 144 respondents, showing 13 operations (in orange in table 1) and asking which *swidgets* users know and use, as well as their perception of the performance of *swidgets*. This survey allows to collect a larger amount of data on which *swidgets* are known and used, as well as detailed information regarding how they were discovered and the reasons that led to the adoption of these controls.

Swidgets exist in various touch-based operating systems, but we decided to focus on iOS and its main applications because it maximizes the odds that participants had a chance to use and know the *swidgets* in these applications before the study, and ensures the set of *swidgets* studied is coherent and stable across application versions and device models², following the same guidelines and logic. This consistent interaction environment, with a variety of hidden controls in it, makes iOS well adapted for conducting first studies on that topic.

²With only minor differences on System *swidgets* of most recent iPhones not equipped with a home button.

Apps		Control Panel				Notification Center			SMS			Mail			
Tasks		Wifi	Flashlight	Screen Orientation	Reveal	Weather	Reveal	Read notification	Delete	Search	See arrival time	Delete	Search	Mark as	Reply
Operations		(un)set	turn on/off	lock to vertical	reveal	consult	reveal	view	from the list	someone	Find time	From the list	someone	read or unread	to a specific mail
Input methods	Swhidgets	System	System	System	System	System	System	System	Item	View	View	Item (+long)	View	Item (+long)	Item
	Navigation Controls	Settings application	Ø	Ø	Ø	Weather application	Ø	Open Apps	Long press	Scroll up and down	Read time of messages	Edit + Delete buttons	Scroll up and down	Flag button	Reply button
	Voice	Yes	Yes	Ø	Ø	Yes	Ø	Yes	Ø	Yes	Ø	Ø	Yes	Ø	Ø

Table 1. List of the possible input methods to complete the operations that participants were asked to perform during the experiments. Depending on the operation, input methods are either a Swhidget, a Navigation control or a Voice control (Ø if input method is not available). Tasks in blue and orange were used in study 1 or study 2, respectively.

STUDY 1: PRELIMINARY STUDY ON AWARENESS AND USAGE OF SWHIDGETS

Design and Procedure

Participants were invited to perform 9 simple *Operations* (summarized in Table 1). Operations can be performed with different *Input methods*: each operation can be performed with one *Swhidget* and one *Navigation* input method, and 3 of them can also be performed with *Voice*³ (using Siri). Operations are grouped into *Tasks*, which order was counterbalanced across participants using a full Latin square, while the order of *Operations* in a task was the same for all participants.

Participants were informed that the goal of the study was to observe interaction on smartphones, without further details. After signing a consent form and answering various demographics questions, they were presented with the study apparatus and tasks. For each of the 9 operations, the experimenter set the iPhone to the home screen and the participant was repeatedly asked to perform the operation until she could not figure a new way to do it. For instance, the operator would first ask *Could you please enable WiFi?* and later ask *Do you know another way to enable WiFi?* The experimenter logged all the input methods used by the participant for this operation (if any) and in which order they were used. As soon as the participant reported not to know any other input method for the prompted operation, the experiment progressed to the next one. Once the participant had completed all 9 operations, the experimenter presented for each of them all the input methods that the participant did not use for this operation, asking if the participant was aware of them and logging the answer. Finally, the experiment concluded with an interview where the participant was invited to fill a questionnaire asking how she discovered the *swhidgets* she used, and if she had the feeling that these controls were useful. An open discussion regarding the study ended the session.

Data Collection

All data were collected manually by the experimenter on dedicated paper forms, and sessions were video-recorded. We

³Note that we considered *Voice* in this study for ecological validity concerns, but it will often ignore it in subsequent analysis when possible to get a complete design

encoded the participant's *degree of knowledge* for each possible combination of *Operation* and *input method* with the following ordinal scale:

1. **PRIMARY** if the participant used that input method first when prompted to perform the given operation.
2. **PRODUCTIVE** if the participant used that input method, but not first, when prompted to perform the given operation.
3. **KNOWN** if the participant did not use that input method when prompted to perform the given operation, but reported to be aware of it during the post-experiment interview.
4. **UNKNOWN** if the participant did not use that input method when prompted to perform the given operation and reported not to be aware of it during the post-experiment interview.

Using this classification, we also consider as **USED** an input method that was used by the participant (regardless of whether it was used first or not); and as **AWARE** an input method that was either **KNOWN** or was **USED** by the participant during the experiment. We considered an operation as *Completed* when the participant used at least one *input method* to complete it, and *Uncompleted* otherwise. i.e. when the participant's degree of knowledge for this operation is **KNOWN** or **UNKNOWN** for all input methods.

Participants and Apparatus

We recruited 28 participants (12 females), with an average age of 39.5 years old (sd=12.6), by sending calls for participation on social networks and mailing lists. All were daily iPhone users for at least 3 months. They had various backgrounds although 13 had a professional activity in IT. For the purpose of the study, participants were provided an iPhone 6S running iOS 11, populated with artificial content to avoid the implication of private data.

Results

Operation completion

Participants could fail or succeed to complete an operation, and a unique **PRIMARY** input method was recorded for this operation only in case of success. We have thus computed for each *Operation* and participant a *primary-or-uncompleted* categorical dependent variable with four levels: *Swhidden*, *Navigation*, *Voice*, and *Uncompleted*. The within-subjects

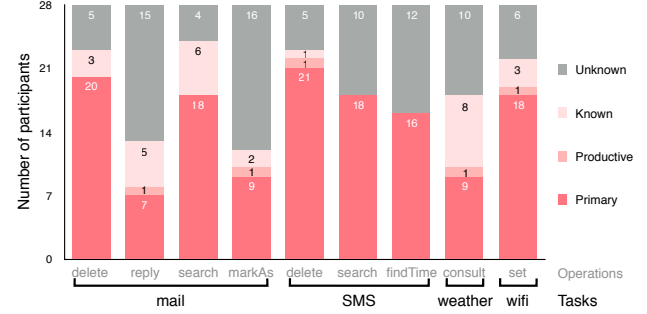
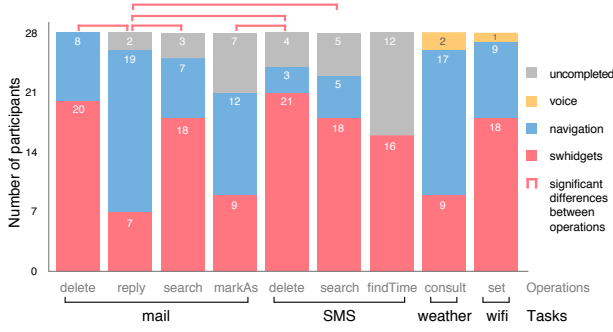


Figure 3. Left: Distribution of participants' primary input methods per operation, with statistically significant differences; Right: Distribution of PRIMARY, PRODUCTIVE, AWARE and UNKNOWN per operation for *swidgets* only.

categorical independent variable is *Operation* (9 levels, see Table 1), and the factorial design is complete. Overall and by-operation distribution of *primary-or-uncompleted* data is shown in Figure 3 (left) – which suggests an important effect of *Operation* on *primary-or-uncompleted*. We analyze this dataset with two Cochran's Q test, one considering if the participants completed or failed each operation, and the other considering if the participants used *swidgets* as PRIMARY or not. Since both tests are performed on the same dataset, we report the Holm-Bonferroni adjustments of the p-values (noted p^* in this paper).

Failure to complete an operation was rather infrequent, with 86.9% of all operations completed. The distribution of the number of *Uncompleted* operations by participant is rather homogeneous, with 20 participants having completed at least all operations but one. *Operation* had a statistically significant effect on *Uncompleted* frequency ($Q=40.5$, $df=8$, $p<10^{-5}$), as could be guessed from Figure 3. Post hoc pairwise comparisons between operations with Wilcoxon signed-rank tests revealed statistically significant differences in operation completion rate only between SMS.findTime and each of mail.delete, mail.reply, weather.consult, and wifi.set.

Awareness and Usage of *Swhidgets*

To investigate the difference in participants' knowledge of *swidgets* and navigation controls, we perform a statistical analysis with two categorical within-subjects independent variables and one ordinal dependent variable in a complete factorial design. The independent variables are *Operation* (9 levels, see Table 1) and *Input method* (2 levels: *Swhidgets* and *Navigation*⁴). The ordinal dependent variable is the *degree of knowledge*, with only three levels (PRODUCTIVE, KNOWN, and UNKNOWN) since we treat PRIMARY as simply PRODUCTIVE to ensure the hypothesis of independence between the two levels of *Input method* (for a participant and task, there can only be one input method labeled PRIMARY). Following Kaptein et al. recommendation to the HCI community [12], statistical analysis of ordinal data was performed with Brunner's and Puri's non-parametric test LD.F2 from the *nparLD* R package [4, 18]. We report test results for the ANOVA-type statistic ATS_{df} , which can be approximated by a $F_{df,\infty}$ distribution for small and moderate sample sizes [4, p. 36].

⁴While collected for the sake of ecological validity, availability and use of *Voice* was too marginal to be considered here.

Factors affecting overall degree of knowledge. Statistically significant effects of *Operation* were observed ($ATS_{5,89}=7.87$, $p<10^{-7}$), but no effect of *Input method* could be found ($ATS_1=1.73$, $p=.18$). However, interaction between these two factors was statistically significant ($ATS_{6,13}=9.443$, $p<10^{-9}$), suggesting that *Input method* had different effect directions depending on the *Operation*. Post-hoc tests conducted with the LD.F1 test and Holm-Bonferroni adjustments indeed revealed a statistically significant effect of *Input method* for SMS.findTime ($ATS_1=18.5$, $p<10^{-4}$, $p^*<10^{-3}$), SMS.delete ($ATS_1=18.4$, $p<10^{-4}$, $p^*<10^{-3}$), mail.reply ($ATS_1=16.1$, $p<10^{-4}$, $p^*<10^{-3}$), and weather.consult ($ATS_1=16.1$, $p<10^{-4}$, $p^*<10^{-3}$). Direction of effect was different in these four operations, with a better knowledge of *swidgets* for SMS.findTime and SMS.delete, but the opposite for mail.reply and weather.consult.

Awareness of swidgets. We investigate the effects of *Input method* and *Operation* on participants' awareness of the existence of an input method to perform an operation, by merging the USED and KNOWN levels of the dependent variable *degree of knowledge* into a single AWARE. On average, participants were aware of 67.1% of the *swidgets* ($SD=24.18\%$) and 70.7% of the navigation controls ($SD=14.9\%$). Awareness of *swidgets* (median=6.5 operations) (shown on Figure 3-right) was relatively similar to the awareness of navigation controls (median=6.07 operations). Statistically significant effect of *Operation* ($ATS_{5,54}=8.02$, $p<10^{-7}$) and of the interaction *Operation* \times *Input method* ($ATS_{6,07}=6.13$, $p<10^{-5}$) were observed, but no effect of *Input method* could be found ($ATS_1=.70$, $p=.40$). Post hoc tests confirmed the effect direction and statistical significance of *Input method* found earlier for mail.reply, SMS.findTime and weather.consult.

Usage of swidgets. We investigate the effects of *input method* and *operation* on participants' ability to effectively use these input methods by merging the PRIMARY and PRODUCTIVE levels of the dependent variable *degree of knowledge* into a single USED. On average, participants used 56.0% of the *swidgets* ($SD=24.7\%$) and 40.9% of the navigation controls ($SD=16.9\%$). Distribution of usage across participants suggests that *swidgets* were more used (median=5.6 operations) than navigation controls (median=3.8 operations), which was confirmed by a statistical effect observed for *Input method* ($ATS_1=4.70$, $p=.030$). Statistically significant effect of *Op-*

eration ($ATS_{6,19}=5.44, p<10^{-5}$) and of the interaction *Operation* \times *Input method* ($ATS_{6,16}=10.42, p<10^{-11}$) were also observed. Post hoc tests confirmed the effect direction and statistical significance of *Input method* for the four operations identified in the previous paragraph, and also revealed an effect toward swhidgets for SMS.search ($ATS_1=8.24, p=.0041, p^*=.021$).

Primary use of Swhidgets. Swhidgets were the most frequent PRIMARY interaction type, with 136 of the 219 completed trials (62.1%), before navigation with 80 (36%), and voice 3 (1%). However, distribution of participants by the number of tasks they completed with a swhidget as PRIMARY was rather unbalanced (min=1, Q1=3.8, median=5, Q3=6, max=9), suggesting that knowledge of –or preference for– swhidgets is quite uneven among our participants. *Operation* had a statistically significant effect on the rate of PRIMARY usage of swhidgets ($Q=37.7, df=8, p<10^{-4}$). Post hoc pairwise comparisons between operations with Wilcoxon signed-rank tests revealed statistically significant differences only between 5 pairs of operations, which are shown in Figure 3 (Left).

STUDY 2: ONLINE SURVEY ON AWARENESS, USAGE AND DISCOVERY OF SWHIDGETS

The results of the previous study suggest that users have an uneven knowledge of *swhidgets*, and that this input method tend to be preferred to other types of interactions. We then conducted a second, larger scale, online study in order to confront the results of the previous study in terms of awareness and usage of *swhidgets*, but also gain more insights on their perceived efficiency, as well as better understand how users discover them.

Design and Procedure

We designed an online survey in order to investigate usage of various *swhidgets*. It started by asking demographic questions, and general questions regarding smartphone/tablet usage and interest in new technologies. Then, for each operation, the respondent was first asked whether she knows how to perform it or not, and if yes, she was asked how frequently this operation was performed. Then, for each associated input method, she was first shown a video demonstrating this input method and then asked whether this input method was new to her. If the input method was unknown, she was asked to report on a 1-7 scale if the input method looked *easy to discover, more convenient than the way usually used to perform the operation*, and if she *would consider adopting this input method in the near future*. If she already knew the input method, she was asked if using it is part of her everyday interaction, if the technique was *easy to discover* (1-7 scale), and if she remembered how she discovered it (for *swhidgets* only). Finally, regardless of the input method, we asked if the respondent would have considered any reasons not to adopt it. This sequence of video and questions was then repeated for each input method corresponding to this operation. Before to move on to the next operation, the respondent was asked to rank all possible input methods by order of preference. Having run through all the *operations*, respondents reach the final part of the survey asking various questions regarding *swhidgets*, among others

about their perceived performance and assessing if the logic behind the physical metaphor was understood.

The survey investigated 13 *operations* across 4 *applications* (see Table 1, orange). Order of the application was counterbalanced across participants, and order of operations was randomized within an application with two exceptions⁵. Order of input methods was the same for all operations. Similarly to study 1, *Voice* control was included for ecological validity but often ignored in the following analysis.

Apparatus and Participants

The survey was created using the LimeSurvey platform [1] and the RemoteControl 2 API. Participants were recruited via calls for participation on social networks. All data were collected anonymously with a limitation of legal age, participation was entirely voluntary and not compensated.

Results

Participants Background and Experience

The survey was completed by 144 respondents (114 females, 29 males, 1 other), aged 18-75 (mean=36.4, SD=10.0). They were relatively educated, with 82.6% having at least an undergraduate-level degree, 74.3% having graduated, and 22.9% owning a master's degree or higher. Participants had various backgrounds, as they described their main working industry using 27 of the 30 different industrial categories. Eight industries concentrated 50.3% of the participants: homemaker (9%), health care and social assistance (6.9%), information services and data processing (6.3%), publishing (6.3%), design (5.6%); arts, entertainment, and recreation (5.6%), software (5.6%), and other information industry (4.9%). Overall, participants reported to be interested in “collecting the latest technology news”, the median response on a 1-7 scale⁶ being 5 (lower quartile: Q1=4, upper quartile: Q3=6), with 59.0% of respondents answering 5 or more, and only 4.9% answering 3 or less. All participants were daily iPhone users and reported to be quite experienced with iOS devices, as 93% used one for at least two years (over 6 years for 48.6%). Their amount of experience using iOS was positively correlated to their age (Spearman's $\rho=.345, p<10^{-4}$). 93.1% of them declared spending at least one hour a day on an iOS device, 67.4% over three hours, and 43.8% over five hours. On a scale from 1=“I know nothing” to 7=“I know everything”, they all evaluated their knowledge of how to interact with their iOS device at 3 or higher: strictly above 3 for 93.1%, and at 5 or more for 74.8% of them (Q1=4, median=5, Q3=6). This score positively correlates with the agreement to “I am highly interested in collecting the latest technology news” ($\rho=.486, p<10^{-9}$), which suggests that following the technology news contributes to participants' knowledge of iOS, or at least to their confidence in knowing it. In the following, we analyse the results regardless of participants' background and experience since they did not reveal any interesting significant effect.

⁵Reveal for Notification Center and Control Panel which were always presented first as they are needed to achieve the other operations.

⁶with 1 being “totally disagree” and 7 “totally agree”

General Usage

Overall knowledge of input methods. Of all 28 input methods presented in this study, participants were aware of 19.7 (70.4%) in average (Q1=15, median=21, Q3=25), knew how to use 18.1 (64.6%) (Q1=14, median=19, Q3=23), and reported to effectively use 13.9 (49.6%) (Q1=10, median=15, Q3=17).

Application and task completion. The four applications used in this study were well known, evaluated by the awareness of at least one input operation to complete them: 96.5% of participants for the Control Panel, 92.4% for Notification Center, 91.0% for Messages and 84.0% for Mail. Participants knew an average of 3.64 applications (SD=0.72), and this number was positively correlated with the self-estimated knowledge of iOS ($p=.34$, $p<10^{-4}$) and interest in technological news ($p=.24$, $p=.0043$). Among all 12 tasks, searching a mail is the one with the smallest proportion of participants who know at least one input method to do it (67.4%), while activating Wi-Fi is the most frequently known task (95.8%). Lack of knowledge of the tasks can be explained only partially by a lack of knowledge of the applications: the proportions of participants who know at least one input method to perform the task among those who know the application range from 77.9% for seeing SMS timings to 99.3% for setting the Wi-Fi (Q1=84.1%, median=92.3%, Q3=94.7%). The 12 tasks had statistically significant differences in reported usage frequency (Skillings-Mack [7] test for ordinal repeated measures with missing values: test statistic = 307.1, $p<10^{-6}$). Based on reported frequency, the most frequent tasks were all notification-related tasks and revealing the Control Center (several times a day), followed by enabling Wi-Fi, locking the screen orientation, and all Mail-related tasks (several times a week), and finally, flashlight and Messages-related tasks (once a week or less).

Awareness and Usage of Swhidgets

Awareness. We evaluate user awareness of a *swhidget* by the proportion of participants stating that the input method was not new to them. Awareness rates of *swhidgets* range from 35.4% (see SMS times) to 95.1% (set Wi-Fi), with a median of 80.2% (Q1=64.0%, Q3=89.1%). Considering only participants who actually knew how to complete the operation, awareness rates of *swhidgets* ranges from 50% to 100% with a median of 95.0% (Q1=89.8%, Q3=99.8%). A repeated measures logistic regression was used to model participants' odds of being aware of an input method (when they know the task it solves), and revealed statistically significant effects of task ($\chi^2=331$, $df=11$, $p<10^{-15}$), input method ($\chi^2=485$, $df=2$, $p<10^{-15}$), and interactions between task and input method type ($\chi^2=84.4$, $df=11$, $p<10^{-12}$). Post-hoc Tukey tests show statistically significant differences between all types of input methods, with average awareness rates for voice (64.4%) being lower ($OR=137$, $z=-4.40$, $p<.001$) than for *swhidgets* (88.6%), themselves lower ($OR=2.37$, $z=2.62$, $p=.020$) than for navigation controls (96.6%). A logistic regression on the *swhidgets* awareness odds alone showed that awareness rates for the three types of *swhidgets* were different ($\chi^2=199$, $df=2$, $p<10^{-15}$). Post-hoc Tukey tests show that awareness rates for view *swhidgets* (78.4%) are lower ($OR=1.69$, $z=-3.074$, $p=.006$) than those for item

swhidgets (80.8%), which are themselves lower ($OR=6.13$, $z=10.5$, $p<.001$) than those for system *swhidgets* (98.3%).

Usage. We evaluate users' usage of a *swhidget* by the proportion of participants stating that they use the corresponding input method or used it in the past. Usage rates are high for most *swhidgets*, ranging from 28.5% (see SMS times) to 84.7% (set Wi-Fi), with a median of 66.0% (Q1=54.3%, Q3=81.6%)⁷. When only considering users who were aware of the *swhidget*, the proportions range from 66.3% to 94.6% with a median of 83.1% (Q1=81.2%, Q3=89.4%). A repeated measures logistic regression modeling participants' odds of knowing an input method they are aware of revealed statistically significant effects of task ($\chi^2=124$, $df=11$, $p<10^{-15}$), input method type ($\chi^2=465$, $df=2$, $p<10^{-15}$), and interactions between task and input method ($\chi^2=47.4$, $df=11$, $p<10^{-5}$). Post-hoc Tukey tests revealed that usage of voice control (29.8%) is significantly lower ($OR=42.0$ and 43.5 , $z=-9.33$ and -7.88 resp., $p<10^{-5}$ for both) than navigation controls (83.4%) and *swhidgets* (86.5%), but the difference between *swhidgets* and navigation control is not significant ($OR=1.04$). A logistic regression on the *swhidgets* alone showed that usage rates for the three types of *swhidgets* were different ($\chi^2=189$, $df=2$, $p<10^{-15}$). Post-hoc Tukey tests show that usage rates for view *swhidgets* (85.1%) are lower ($OR=1.71$, $z=-2.97$, $p=.0084$) than those for item *swhidgets* (80.7%), which are themselves lower ($OR=5.64$, $z=10.6$, $p<.001$) than those for system *swhidgets* (90.5%)⁸.

Preferred input method. Unlike in previous study, we could not query which input method is spontaneously used first for performing an operation. Instead, we asked participants to rank by order of preference all possible input methods for each operation. The proportions of participants aware of a *swhidget* for an operation who ranked it as their preferred input method range from 39.5% (delete mails) to 91.5% (flashlight), with a median of 68.6% (Q1=53.6%, Q3=72.3%). These results suggest that users tend to prefer *swhidgets* when they are aware of them, but that it also depends on the operation. For instance, several respondents reported to use conventional buttons for deleting e-mails because it is more convenient, as they often delete them right after reading and using the *swhidget* would require to come back to the list first. Note that among respondents who were not aware of the *swhidget* for an operation, proportions range from 18.2% to 85.7% (median=40.0%, Q1=25.0%, Q3=53.6%), showing that participants could also be highly interested in discovering some of the *swhidget* they do not know. As illustrated on Figure 4-Left, participants preferred *swhidgets* for 6 tasks out of 9. It was preferred over Navigation methods 5 times out of 8 when both input methods were available. Voice was by far the less preferred input method, which is not surprising given that respondents were also much less aware of it.

⁷Note that for 22 respondents out of 144, a database issue prevented us to retrieve the answer to this question for the *swhidget* of the Messages search operation.

⁸These proportions are not in the order revealed by the logistic regression because the latter accounts for variations in the knowledge of participants, and therefore reduces the weight of those who know more input methods. On the other hand, the proportions given are biased toward the behavior of the most knowledgeable participants.

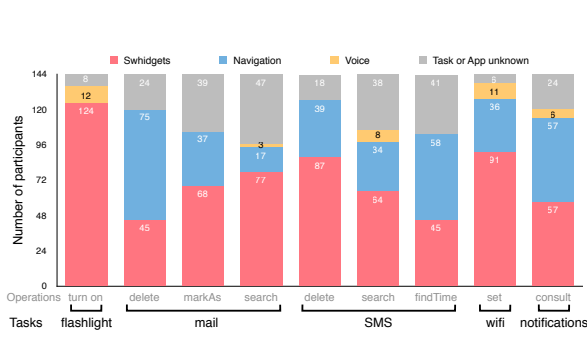
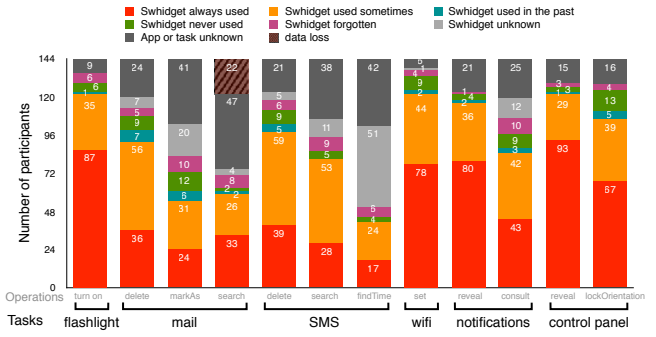


Figure 4. Left: Distribution of preferred input techniques to perform operations, when alternatives to *swhidgets* exist; Right: Distribution of usage frequency of *swhidgets*.



Discovery of Swhidgets

On all participants and operations, exploration of the interface was the most frequently reported source of discovery of swhidgets (42.6%), followed by accidental activation (15.9%), transfer of knowledge from another application or system (8.8%), explicitly looking or asking for it (8.6%), informing herself generally about the interface or application (6.3%), being demonstrated the input method (4.9%), remembering seeing someone else using it (3.1%), and others (1.0%). Surprisingly, respondents replied “I cannot remember” for only 8.8% of the *swhidgets*. That being said, it would be presumptuous to over-interpret these results, as it remains uncertain that participants remember well how they learned a input method, especially since our participants have a long experience of iOS and probably learned the *swhidgets* a long time ago. As such, it is likely that they assumed how they discovered them, rather than how they actually did discover them.

Swhidgets’ physical metaphor. There was generally no consensus on the questions “I can tell which objects could be moved to reveal *swhidgets*” (median=4, Q1=2, Q3=5) and “I can tell in which direction I should move objects to reveal *swhidgets*” (median=4, Q1=3, Q3=5). These two ratings were strongly correlated ($\rho = .709$, $p < 10^{-22}$), which suggests that the design of *swhidgets* in iOS allows users to integrate these two aspects of the interaction in a single concern. The consistence of the design of *swhidgets* in iOS interactions was better acknowledged, however, with 61.1% of participants agreeing (scored 5-7) and only 8.3% disagreeing (scored 1-3). Although there was no consensus on the statement “I do not expect controls to hide under interface elements or bezels”, 45.8% respondents disagreed (scored 1-3) whereas 22.2% agreed (scored 5-7). Agreement with this statement was slightly and negatively correlated with the interest for technologies ($\rho = -.226$, $p = .0064$), which suggest that despite being rooted in real world physics, additional knowledge might be necessary to grasp this aspect of the metaphor.

Transfer of Knowledge Through Application. There was a clear consensus among participants to agree with the statement “When I know a specific *swhidget* in an application, I suspect other places with a similar screen layout to provide similar *swhidgets*.” (Q1=4, median=6, Q3=7). Most participants agreed that when they discover a *swhidget*, they “play [with it] to find all the possible ways to use it” (Q1=4

median=5, Q3=7). Self-reported knowledge of iOS interaction was slightly correlated to this item’s answers ($\rho = .283$, $p < .001$), which suggests that users who “play” with *swhidgets* to explore interaction possibilities have more chance to know about iOS interaction or, at least, to feel more confident about their knowledge. Participants are divided on the idea that having discovered a *swhidget*, they should expect it can be swiped in another direction to reveal other functions (38.2% disagree, 31.9% disagree, median=4, Q1=2, Q3=5). This is not necessarily surprising considering that *swhidgets* are not systematically symmetric. Typically, lists in many applications have a delete *swhidget* on the right of the items but not necessarily have *swhidgets* on the left.

Mental Models and the Logic of Swhidgets. We asked participants to evaluate their agreement with seven statements concerning “how the placement and type of *swhidgets* relate to the functions they provide [and if participants could] find this logic and make it [theirs]”. There was a global agreement (Q1=4, median=5, Q3=6) for the following six statements: 1) participants understand the logic prevailing to the different types of *swhidgets* and their placement in the interface; 2) they can use this logic to guide their search for a *swhidget* with the function they need; 3) they sometime feel that different parts of an application failed to follow a common logic for their *swhidgets*; 4) they notice *swhidgets* that follow a similar logic in different applications or systems; 5) once they learned how to do something with a *swhidget*, they cannot think anymore about another way to do it; 6) they sometime forget that there are *swhidgets* they can use to do a task. As one can expect, there is a strong correlation between items 1 and 2 ($\rho = .787$, $p < 10^{-30}$), which were both correlated with participants’ self-estimated knowledge of iOS ($\rho = .279$ and $.217$, $p^* = .0043$ and $.047$ respectively) and interest for technology news ($\rho = .268$ and $.308$, $p^* = .0012$ and $.0002$ respectively).

General perception of Swhidgets

Interacting with Swhidgets. Participants appreciate interacting with *swhidgets*, reporting that they enjoy it (Q1=5, median=6, Q3=7, 85.3% scored 5-7) and that the interface responds in a way that feels satisfactory (Q1=5, median=6, Q3=7, 88.1% scored 5-7). Participants globally rejected the difficulty of manipulating *swhidgets*, at 82.5% for “they are difficult to perform”, 64.3% for “the system often misinterprets my swipe gestures”, and 62.2% for “their animation is sometimes unpre-

dictable”. There was nonetheless respectively 9.8%, 15.4% and 13.3% of participants who agreed with these statements.

Reasons Not to Adopt Swhidgets. The main reason reported for not adopting a *swhidget* was *habits* (“It is unadapted to my habits and/or preferences”, 38.6% of all evaluations), followed by *context* (“It is unadapted to my contexts of use”, 26.6%), *perform* (“It is difficult to perform”, 14.5%), *worth* (“It is not worth changing my habits”, 11.2%), and *learn* (“It is difficult to learn”, 5.4%). Distinguishing the three types of *swhidgets* revealed a statistically significant difference for *context* ($p=.043$), with system *swhidgets* having respectively 1.37, and 1.38 times greater odds than view *swhidgets* ($p=.044$), item *swhidgets* ($p=.010$). The few open comments regarding reasons not to adopt varied considerably from one respondent to another. As an example, one respondent reported that she had difficulties revealing the Notification Center, as her swipe was often not recognized by the system, which changes her preference of this *swhidget*, whereas as seen above, the statement “the system often misinterprets my swipe gestures” was in majority rejected by respondents. Interestingly, even though it was not asked in the survey, several participants insisted for some *swhidgets* that they did not see any reason not to adopt them (reported by 13 respondents for *flashlight* and 11 for locking orientation through the control panel).

DISCUSSION

We reported on the results of two studies conducted on current usage of *swhidgets*. The first study conducted in a laboratory setting investigated which *swhidgets* were spontaneously used by participants when prompted to perform certain operations on an iOS device. The second study conducted via an online survey platform, investigated which *swhidgets* users reported to know and use. Combined, our studies provide the following main insights on awareness, usage and discovery of *swhidgets* by middle-aged and technology-friendly users.

Awareness of Swhidgets

Awareness of *swhidgets* was of 64.1% and 80.2%, in study 1 and 2 (respectively). While unsurprisingly lower than awareness of navigation controls, awareness rate of our participants was relatively high, yet slightly different between both studies. One possible explanation of this difference may be the fact that it was *observed* in study 1 and *self-reported* in study 2. Regardless, these results suggest that the integration of *swhidgets* in touch-based operating systems, despite the lack of signifier to inform users of their availability, is well executed and that users are still able to discover their existence. Interestingly though, awareness rate also happened to vary depending on the operation. Notably, 42.9% and 50% of participants in study 1 and 2, respectively, were not aware of the view *swhidget* for revealing the time at which a SMS was sent in Message. It is of particular interest since there is no real alternative to obtain this information, and that messaging was one of the main activity reported to be conducted by our participants with their smartphones. Unaware participants provided mixed feedback about this feature when this item *swhidget* was demonstrated to them, either commenting that they were simply not aware of it where others commented that they never need to know the exact time of a message. Awareness also depends on the type

of *swhidgets*, with users significantly more aware of *system* than *item* and *view swhidgets*. In the end, similarly to hotkeys that are more often learned for frequent commands [13, 14], it is likely given their complete lack of signifier that users are more aware of *swhidgets* they feel the need to use, which probably motivates their discovery in the first place.

Usage of (And Reasons Not to Use) Swhidgets

Usage of *swhidgets* was on average of 56.0% and 66.0%, in study 1 and 2, respectively (once again, usage rate was lower for *swhidgets* than for navigation controls, as one could expect). Overall usage rates are relatively high, especially considering awareness rates as they show that our participants tend to use a high proportion of the *swhidgets* they are aware of (median usage rate of 83.1% in study 2, up to 94.6%). Even more interesting, we observed that *swhidgets* are extremely often favored over alternative inputs. In study 1, 62.1% of all completed trials were performed using first a *swhidget*, with 36% only for navigation. In study 2, *swhidgets* were preferred on average by 57.8% of participants who knew how to complete the operation, against 39.2% only for navigation controls (*swhidgets* favored over navigation for 5 operations). This result can once again be explained by the fact that *swhidgets* may appear as a *shortcut* alternative to navigation controls in many applications, and as such, are favored over techniques that would take longer or would be more tedious to use. However, proportion of primary or preferred usage once again depends on the operation performed. Typically, in study 2, only 37.5% of participants who completed the *delete e-mail* operation preferred the *swhidget* over the navigation methods. Similarly, in study 1, only 36% preferred the *swhidget* for replying to an e-mail. The main reasons reported were that these operations are often performed right after reading the e-mail, where conventional navigation controls are ready-to-hand, whereas using the *swhidget* would require to come back to the e-mail list. Once again, these results suggests that usage is also likely to be higher for *swhidgets* that appear as shortcut alternative to conventional input methods.

In study 1, 21.4% of participants did not consider that *swhidgets* improve their everyday interaction. We further investigated the reasons behind this in study 2, where 38.6% of respondents reported that the main reason not to adopt a *swhidget* would be because it is “unadapted to [their] habits”, and 26.6% because it is “unadapted to [their] context of use”, confirming the assumptions made above about *swhidgets* considered mostly useful as shortcuts. However, it is known that users have difficulties to estimate the actual costs and benefits of switching to a theoretically more efficient alternative [8, 14]. This was confirmed by our observations, where several users had a wrong perception of the performance gain provided by *swhidgets*. For instance, in study 1, several participants who knew how to delete emails with an item *swhidgets* still preferred to tap on the visible button, because they believe it is faster to select all the emails that they want to delete and then put them into the trashcan together. Although this is only our perception that *swhidgets* would be faster in this case, and that they are indeed designed to be faster, an implication for design would be to increase the visibility of this fact so that the speed difference is more accurately perceived by users.

Discovery of Swhidgets

Study 2 highlighted the role of self-exploration of the interface in the discovery of *swhidgets*, where we asked how users discovered each *swhidgets*. 42.6% of the answers were that the *swhidget* was discovered by exploring the interface, and 15.9% that it was from accidental activation. These results are in line with the assumptions of Schramm et al. [26], that users become more familiar with touch-based devices and *swhidgets*, and tend to explore the interface to discover more of them.

Schramm et al.'s assumptions are also in line with the results collected in study 2 regarding the understanding of the physical metaphor of *swhidgets*. While we found no consensus regarding a complete understanding of which objects could be moved to reveal *swhidgets*, respondents clearly reported that they suspect to find *swhidgets* they know at similar location in different applications, which probably encourages interface exploration, and therefore, the discovery of *swhidgets* that way. That being said, our results suggest that knowledge of *swhidgets* transfers only in limited amounts from one application to another, between *swhidgets* that rely on similar interface layouts. For instance, the similitude of participants' knowledge about deleting mails and SMS with *swhidgets* (in both studies), as well the similitude of PRIMARY usage of *swhidgets* for searching mails and SMS in study 1 are compatible with such a transfer. However, the significant difference between PRIMARY usage of SMS.delete and mail.markAs – which use basically the same kind of *swhidgets* – suggests that the transfer of knowledge does not rely only on the type of *swhidgets* involved, but also on the nature of the task and direction of the swipe gestures (participants of study 2 differed in opinions about expecting different swipe directions).

Finally, the low numbers of *swhidgets* discovered following a demonstration of the input method (4.9%) or remembering seeing someone else using it (3.1%) was however surprising. Indeed, social interaction had been found to play a major role in disseminating useful system capabilities [23]. Similarly to usage of *swhidgets*, it is possible that users have a wrong estimation of the performance benefits of a *swhidget* when witnessing someone using it. In addition, it is possible that our respondents discovered the *swhidgets* months or years before completing the survey, therefore it remains uncertain that participants remembered perfectly how they did discover them. Even though a “I do not remember” option was available, it is likely that users sometimes preferred to assume how they discovered them rather than answering they did not remember.

Future work

While informative, our studies could not explore all smartphone ecosystems, as well as focus on specific populations. As such, it leaves interesting opportunities for future work.

First, we chose to focus our studies on the iOS ecosystem for several reasons: it is a coherent and consistent OS, whereas Android allows manufacturer to create software overlays that change the look and feel of the OS. Because of this, default applications, advanced input methods (including *swhidgets*), hardware inputs, etc. are likely to differ much more between Android devices than between iOS devices. That being said, Apple recent smartphones and tablets are not equipped with a

hardware home button anymore. The home feature is now performed with an upward swipe from the bottom of the display, which interferes with the previously control panel *swhidget*. Consequently, on these devices, downward swipes from the top edge are now used both for revealing Control Panel and Notification Center, depending on which corner they are performed from. Our studies did not investigate these novel gesture mappings for this system *swhidget*, but remain valid for all other *swhidgets* tested. Future work should therefore focus on investigating *swhidgets* on other platforms, and possibly investigating the discovery and awareness of the localized *swhidgets* on recent iOS devices.

Second, even though we recruited participants of various ages (18 to 75 years old), both of our studies were conducted in majority with middle-aged adults familiarized with novel technologies. As such, it is likely that our participants, while representative of many smartphone users, correspond to a relatively high limit in term of knowledge of interactive capabilities of their device and interest in latest technologies. It is unlikely that this limited pool of participants would diminish in any way the validity of our results, yet future work should investigate different populations. While we informally tested the effect of age on awareness and usage of *swhidgets* but did not find any strong effect, we still believe that future work should investigate this question with different age groups. Seniors and teenagers have significantly different backgrounds and contexts of usage of technologies that it would be interesting to investigate the impact of these differences.

Dataset

Data collected for this work has been anonymized and is available to the community through our institution address⁹. We collected a significant amount of information in order to compare interaction with *swhidgets* with navigation and voice control, but focused our analysis on *swhidgets* which is the core of this paper. That being said, we remind that this dataset contains information regarding many of the points discussed (typically regarding discovery, reasons to adopt, etc.) for navigation and voice controls as well.

CONCLUSION

The results of our two studies show that participants have a moderately high awareness of iOS *swhidgets*, use most of those they are aware of, and often prefer *swhidgets* over alternative input methods when their performance benefits are clear. While relying on a signifier-less design, *swhidgets* are relatively well discovered by users through exploration of the interface. As such, touch-based OSES could possibly offer more of these controls to the users. However, the fact that discovery happens mostly through self-exploration also suggests that users do not necessarily rely on the online resources provided by manufacturers, possibly leaving room for an improved design that would ingeniously signify these controls to the user from time to time during the interaction. This work is to the best of our knowledge the first study conducted on this type of mainstream User Interface components.

⁹<http://ns.inria.fr/discovery/swhidgets/>

REFERENCES

- [1] 2019. Lime Survey. (2019). <https://www.limesurvey.org/> Retrieved July 8, 2019.
- [2] Apple. 2019. Apple Human Interface Guidelines. (2019). <https://developer.apple.com/design/human-interface-guidelines/ios/overview/themes/> Retrieved July 8, 2019.
- [3] Jeff Avery and Edward Lank. 2016. Surveying Expert-Level Gesture Use and Adoption on Multi-Touch Tablets. In *Proceedings of the 2016 ACM Conference on Designing Interactive Systems (DIS '16)*. ACM, New York, NY, USA, 577–581. DOI: <http://dx.doi.org/10.1145/2901790.2901895>
- [4] Edgar Brunner and Madan L. Puri. 2001. Nonparametric methods in factorial designs. *Statistical Papers* 42, 1 (01 Jan 2001), 1–52. DOI: <http://dx.doi.org/10.1007/s003620000039>
- [5] Bill Buxton, Sheelagh Carpendale, Saul Greenberg, and Nicolai Marquardt. 2012. *Sketching user experiences*. Morgan Kaufmann.
- [6] John M Carroll and Mary Beth Rosson. 1987. *Paradox of the active user*. The MIT Press.
- [7] Mark Chatfield and Adrian Mander. 2009. The Skillings–Mack test (Friedman test when there are missing data). *The Stata Journal* 9, 2 (2009), 299–305.
- [8] Andy Cockburn, Carl Gutwin, Joey Scarr, and Sylvain Malacria. 2014. Supporting Novice to Expert Transitions in User Interfaces. *Comput. Surveys* 47, 2, Article 31 (Nov. 2014), 36 pages. DOI: <http://dx.doi.org/10.1145/2659796>
- [9] Clarisse Sieckenius De Souza. 2005. *The semiotic engineering of human-computer interaction*. MIT press.
- [10] Computer Hope. 2019. Definition of Charms. (2019). <https://www.computerhope.com/jargon/c/charms.htm> Retrieved July 8, 2019.
- [11] Mohit Jain and Ravin Balakrishnan. 2012. User Learning and Performance with Bezel Menus. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '12)*. ACM, New York, NY, USA, 2221–2230. DOI: <http://dx.doi.org/10.1145/2207676.2208376>
- [12] Maurits Clemens Kaptein, Clifford Nass, and Panos Markopoulos. 2010. Powerful and Consistent Analysis of Likert-type Ratingscales. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '10)*. ACM, New York, NY, USA, 2391–2394. DOI: <http://dx.doi.org/10.1145/1753326.1753686>
- [13] David M Lane, H Albert Napier, S Camille Peres, and Aniko Sandor. 2005. Hidden costs of graphical user interfaces: Failure to make the transition from menus and icon toolbars to keyboard shortcuts. *International Journal of Human-Computer Interaction* 18, 2 (2005), 133–144.
- [14] Sylvain Malacria, Joey Scarr, Andy Cockburn, Carl Gutwin, and Tovi Grossman. 2013. Skillometers: Reflective Widgets That Motivate and Help Users to Improve Performance. In *Proceedings of the 26th Annual ACM Symposium on User Interface Software and Technology (UIST '13)*. ACM, New York, NY, USA, 321–330.
- [15] Sven Mayer, Lars Lischke, Adrian Lankswert, Huy Viet Le, and Niels Henze. 2018. How to Communicate New Input Techniques. In *Proceedings of the 10th Nordic Conference on Human-Computer Interaction (NordiCHI '18)*. ACM, New York, NY, USA, 13. DOI: <http://dx.doi.org/10.1145/3240167.3240176>
- [16] Jakob Nielsen. 1994. *Usability engineering*. Elsevier.
- [17] Jakob Nielsen. 2010. iPad Usability: First Findings From User Testing. (2010). <http://www.useit.com/alertbox/ipad.html> Retrieved September 1, 2019.
- [18] Kimihiro Noguchi, Yulia Gel, Edgar Brunner, and Frank Konietzschke. 2012. nparLD: An R Software Package for the Nonparametric Analysis of Longitudinal Data in Factorial Experiments. *Journal of Statistical Software, Articles* 50, 12 (2012), 1–23. DOI: <http://dx.doi.org/10.18637/jss.v050.i12>
- [19] Donald Norman. 2004. *Emotional design : why we love (or hate) everyday things*. Basic Books, New York.
- [20] Donald Norman. 2013. *The Design of Everyday Things*. Basic Books.
- [21] Donald A. Norman. 2010. Natural User Interfaces Are Not Natural. *Interactions* 17, 3 (May 2010), 6–10. DOI: <http://dx.doi.org/10.1145/1744161.1744163>
- [22] Donald A. Norman and Jakob Nielsen. 2010. Gestural Interfaces: A Step Backward in Usability. *Interactions* 17, 5 (Sept. 2010), 46–49. DOI: <http://dx.doi.org/10.1145/1836216.1836228>
- [23] S. Camille Peres, Franklin P. Tamborello, Michael D. Fleetwood, Phillip Chung, and Danielle L. Paige-Smith. 2004. Keyboard shortcut usage: The roles of social factors and computer experience. In *Proceedings of the Human Factors and Ergonomics Society Annual Meeting*, Vol. 48. SAGE Publications Sage CA: Los Angeles, CA, 803–807. DOI: <http://dx.doi.org/DOI:10.1177/154193120404800513>
- [24] Adhithya Ramakumar. 2017. An analysis of the interactions on iOS 11. (2017). <https://uxdesign.cc/an-analysis-of-the-interactions-on-ios-11-2abc072d42ed> Retrieved September 1, 2019.
- [25] Volker Roth and Thea Turner. 2009. Bezel Swipe: Conflict-free Scrolling and Multiple Selection on Mobile Touch Screen Devices. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '09)*. ACM, New York, NY, USA, 1523–1526. DOI: <http://dx.doi.org/10.1145/1518701.1518933>

- [26] Katherine Schramm, Carl Gutwin, and Andy Cockburn. 2016. Supporting Transitions to Expertise in Hidden Toolbars. In *Proceedings of the 2016 CHI Conference on Human Factors in Computing Systems (CHI '16)*. ACM, New York, NY, USA, 4687–4698. DOI: <http://dx.doi.org/10.1145/2858036.2858412>
- [27] Marcos Serrano, Eric Lecolinet, and Yves Guiard. 2013. Bezel-Tap Gestures: Quick Activation of Commands from Sleep Mode on Tablets. In *Proceedings of the SIGCHI Conference on Human Factors in Computing Systems (CHI '13)*. ACM, New York, NY, USA, 3027–3036. DOI: <http://dx.doi.org/10.1145/2470654.2481421>
- [28] Ben Shneiderman. 1997. *Designing the User Interface: Strategies for Effective Human-Computer Interaction* (3rd ed.). Addison-Wesley Longman Publishing Co., Inc., Boston, MA, USA.
- [29] Susanne Tak, Piet Westendorp, and Iris Van Rooij. 2013. Satisficing and the use of keyboard shortcuts: being good enough Is enough? *Interacting with computers* 25, 5 (2013), 404–416.
- [30] Daniel Wigdor and Dennis Wixon. 2011. *Brave NUI world: designing natural user interfaces for touch and gesture*. Elsevier.